



Υπηρεσία παρακολούθησης θερμοκρασίας σε Server Rooms

Αναστασιάδης Παναγιώτης 22101
Καραβιβέρη Αλεξάνδρα 22103



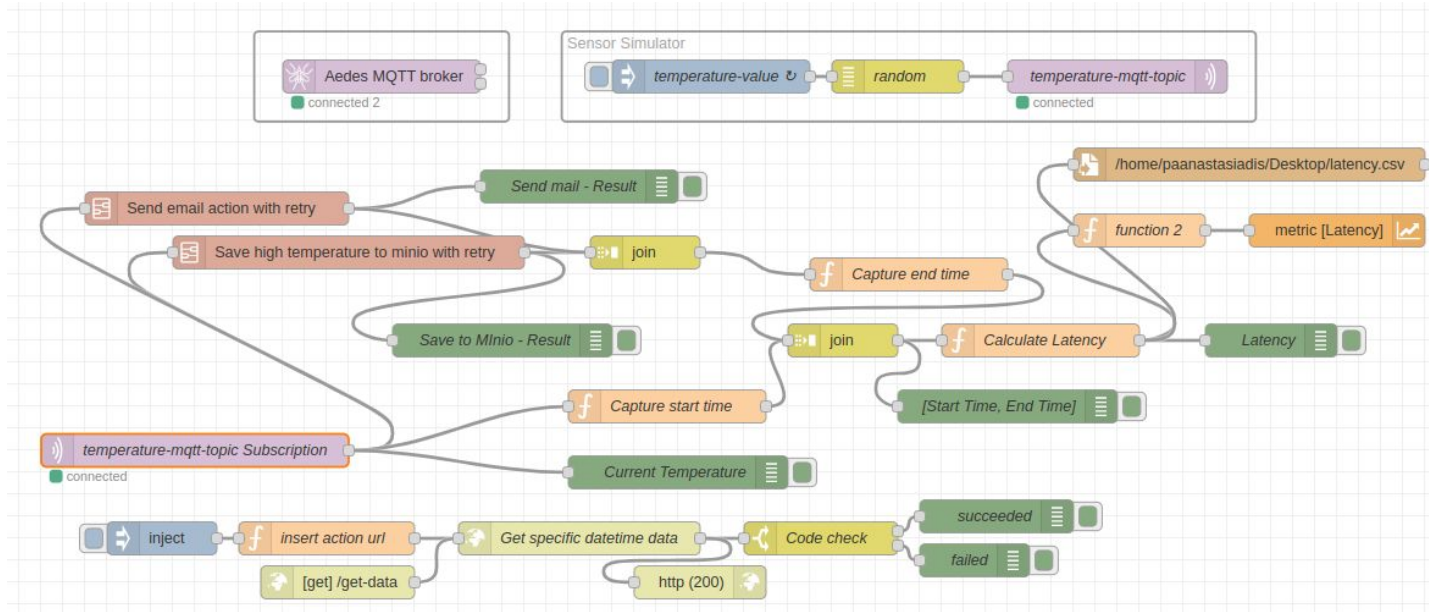
Εισαγωγή

Κρίσιμος ρόλος των data centers: Τα data centers αποτελούν ζωτική υποδομή για οργανισμούς, καθώς φιλοξενούν server rooms απαραίτητα για τη λειτουργία των υπηρεσιών τους.

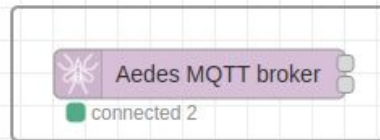
Σημασία θερμοκρασίας: Η διατήρηση σταθερής θερμοκρασίας στους χώρους των data centers και κατά συνέπεια στο hardware τους είναι πολύ σημαντική για την συνεχή διαθεσιμότητα των υπηρεσιών, αποτρέποντας βλάβες οι οποίες οδηγούν σε διακοπές λειτουργίας.

Σύγχρονη λύση στο νέφος: Η εργασία παρουσιάζει μια σύγχρονη λύση που εκμεταλλεύεται τις υπηρεσίες του νέφους και μοντελοποιεί αποτελεσματικά τη διαχείριση της θερμοκρασίας στα data centers, ενισχύοντας την αξιοπιστία των υπηρεσιών.

Αρχιτεκτονική της εφαρμογής



Αρχιτεκτονική (1/4)



Node Red Core: Ο πυρήνας της εφαρμογής βασίζεται στο Node Red, το οποίο διαχειρίζεται και συνδέει τα διάφορα services που αποτελούν την εφαρμογή.

Aedes MQTT Broker Node (Pub/Sub Pattern): Χρησιμοποιείται ο Aedes MQTT Broker σε MQTT επικοινωνία για τη διαχείριση των καναλιών και την επικοινωνία, εγκαθιστώντας τον πάνω στον Node Red server.

Temperature Sensor Simulator (Pub/Sub Pattern): Αναπαράγει τη λειτουργία ενός hardware component που παράγει τιμές θερμοκρασίας και δημοσιεύει τις τιμές αυτές στο MQTT topic "/temperature" για προσομοίωση.



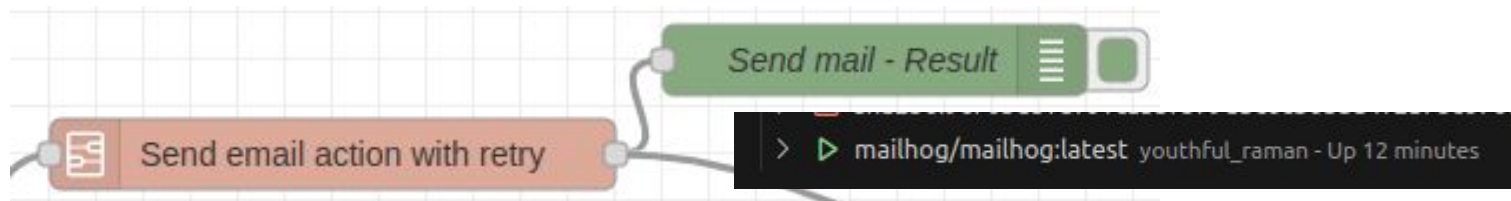
Αρχιτεκτονική (2/4)



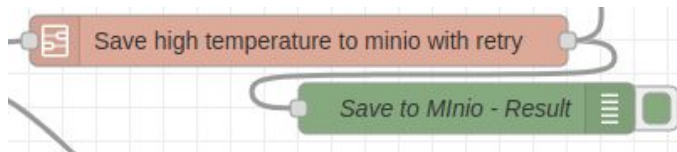
Subscriber Node (Pub/Sub Pattern): Χρησιμοποιείται ως subscriber node για την λήψη και προώθηση των τιμών θερμοκρασίας από το MQTT topic `"/temperature"`.

Sending Mail Alert | Openwhisk Action: Υλοποιεί τη λειτουργία αποστολής email ειδοποίησης σε περίπτωση υψηλών θερμοκρασιών, χρησιμοποιώντας το Openwhisk.

Mailhog Service: Χρησιμοποιείται το Mailhog service για τον έλεγχο της αποστολής και λήψης των emails κατά τη διαδικασία του τεστ και της ενεργοποίησης του Openwhisk action.



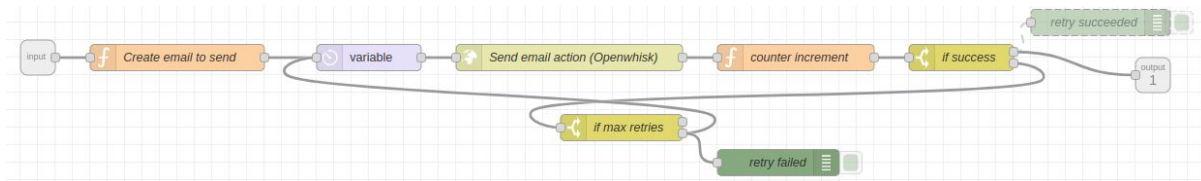
Αρχιτεκτονική (3/4)

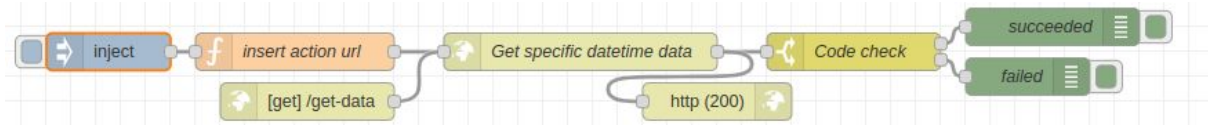


Saving Measurements to Minio | Openwhisk Action: Χρησιμοποιείται το MinIO για την αποθήκευση των μετρήσεων των υψηλών θερμοκρασιών, με την ενεργοποίηση του Openwhisk action.

MinIO: Το Minio αποτελεί Object Storage Service και η χρησιμοποίηση του μας βολεύει έτσι ώστε μελλοντικά να προστεθεί επιπλέον λειτουργία όπως η αποθήκευση εικόνων heatmaps.

Retrying Unsuccessful Openwhisk Actions (Retry Pattern): Υποστηρίζει τη διαδικασία επανάληψης ανά ορισμένα χρονικά διαστήματα των Openwhisk actions σε περίπτωση αποτυχίας.

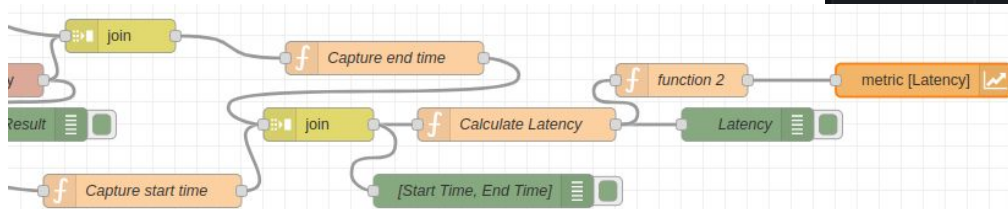




Αρχιτεκτονική (4/4)

REST API | Fetching Data | Openwhisk Action: Υλοποιεί ένα REST API endpoint `/get-data` για την εύκολη ανάκτηση των δεδομένων από το MinIO, με δυνατότητα φιλτραρίσματος βάσει χρονικού διαστήματος.

Monitoring Latency and MQTT Topics | Prometheus and Grafana: Υποστηρίζει τη μέτρηση της καθυστέρησης της βασικής λειτουργίας και την προσκόλληση των MQTT topics μέσω του Prometheus και του Grafana για παρουσίαση των μετρήσεων.





Μοτίβα Σχεδιασμού και Μηχανισμοί

Μοντέλο Pub/Sub: Η εφαρμογή χρησιμοποιεί το μοντέλο σχεδιασμού Pub/Sub με το πρωτόκολλο MQTT για την αποτελεσματική συλλογή μετρήσεων θερμοκρασίας, προσφέροντας αδιάλειπτη επικοινωνία, πραγματικό χρόνο μετάδοσης δεδομένων και ευελιξία στην επέκταση της εφαρμογής.

Πρότυπο Retry: Οι λειτουργίες Openwhisk υπόκεινται σε πρότυπο επανάληψης, διασφαλίζοντας αξιοπιστία και ανοχή σε σφάλματα, καθώς επαναλαμβάνονται σε περίπτωση αποτυχίας, προστατεύοντας την εφαρμογή από απώλεια ειδοποιήσεων και δεδομένων.

Παρακολούθηση και Αυτοδιαχείριση: Χρησιμοποιώντας τα εργαλεία Grafana και Prometheus, η εφαρμογή επιτρέπει συνεχή παρακολούθηση της απόδοσης και κατάστασης λειτουργίας, καθώς καταγράφει τους χρόνους απόκρισης και εντοπίζει διακυμάνσεις. Επιπλέον, οι μηχανισμοί επανάληψης του προτύπου Retry προσφέρουν αυτοδιαχείριση χωρίς ανθρώπινη παρέμβαση.



Πλεονεκτήματα

Αποθήκευση Δεδομένων στο MinIO: Η αποθήκευσης δεδομένων στο MinIO επιτρέπει, μελλοντικά, σε ιστορική ανάλυση της θερμοκρασίας. Αυτό θα μας δώσει τη δυνατότητα να την εντοπισμό τάσεων και απειλών, τη βελτιστοποίηση στρατηγικών διατήρησης σταθερής θερμοκρασίας και τη συμμόρφωση με κανονισμούς.

Συνεχής και Αυτοματοποιημένη Παρακολούθηση Θερμοκρασίας: Η συνεχής παρακολούθηση της θερμοκρασίας σε πραγματικό χρόνο επιτρέπει τη βελτιστοποίηση της κατανάλωσης ενέργειας, μείωση του κόστους και του περιβαλλοντικού αποτυπώματος.



Μειονεκτήματα

Πολυπλοκότητα Εγκατάστασης: Η αρχική υλοποίηση έχει μεγάλη πολυπλοκότητα στην εγκατάσταση, ρύθμιση και διαμόρφωση των διαφόρων components όπως αισθητήρες, Node-RED, MinIO και OpenWhisk.

Υψηλό Κόστος Ενσωμάτωσης και Διατήρησης: Η ενσωμάτωση και η συντήρηση αυτής της αρχιτεκτονικής μπορεί να συνεπάγεται υψηλό κόστος, ειδικά όσον αφορά τους αισθητήρες και την ενεργειακή κατανάλωση. Αυτό το κόστος μπορεί να αυξηθεί περαιτέρω αν χρειαστεί να γίνουν μελλοντικά επεκτάσεις. Επιπλέον, η πιστοποίηση για τη χρήση εργαλείων και υπηρεσιών μπορεί να προσθέσει επιπλέον κόστος.



Σύγκριση με σύγχρονες αρχιτεκτονικές

Πιο Ακριβής Έλεγχος Θερμοκρασίας: Σε σύγκριση με τα παραδοσιακά HVAC συστήματα, αυτή η αρχιτεκτονική προσφέρει πιο ακριβή έλεγχο της θερμοκρασίας, με αποτέλεσμα την εξοικονόμηση ενέργειας. Τα συστήματα HVAC είναι συχνά λιγότερο granular με αποτέλεσμα να μην ανταποκρίνονται τόσο γρήγορα στις διακυμάνσεις της θερμοκρασίας.

IoT-Based Λύσεις: Υπάρχουν παρόμοιες IoT-Based λύσεις, οι οποίες όμως ενδέχεται να μην περιλαμβάνουν τις severless υπολογιστικές δυνατότητες του OpenWhisk, σε αντίθεση με τη συγκεκριμένη αρχιτεκτονική που γίνεται triggered όποτε χρειάζεται με αποτέλεσμα να είναι πιο αποδοτική.

SLA

column0	Rec
6,006	1
6,004	1
6,004	1
6,004	1
6,003	1
6,002	1
6,002	1
6,002	1
6,002	1
6,001	1
6,001	1
6,001	1
6,001	1
6,000	1
6,000	1
6,000	1
5,997	1
5,997	1
5,907	1
5,905	1
5,904	1
5,903	1
5,903	1

latency.csv

Χρόνος Ανταπόκρισης: Στόχος είναι η διατήρηση ενός χρόνου ανταπόκρισης λιγότερο από 6,004 δευτερόλεπτα (99th percentile) για κρίσιμες λειτουργίες, κατά μέσο όρο ανά μήνα, με μέτρηση από την έναρξη της λειτουργίας έως τη λήψη της απάντησης.

Διαθεσιμότητα: Στόχος είναι η διατήρηση διαθεσιμότητας τουλάχιστον 99,9%, με μέτρηση του ποσοστού του χρόνου κατά τον οποίο η υπηρεσία είναι λειτουργική και προσβάσιμη για τον χρήστη, κατά μέσο όρο ανά μήνα.

Διατήρηση Δεδομένων Θερμοκρασίας: Στόχος είναι η διατήρηση ιστορικών δεδομένων θερμοκρασίας για τουλάχιστον 1 έτος, με μέτρηση της διαθεσιμότητας των ιστορικών δεδομένων κατά μήνα.

Ειδοποιήσεις Θερμοκρασίας: Στόχος είναι η έκδοση ειδοποιήσεων θερμοκρασίας όταν αυτή υπερβαίνει τους 30°C, με κατώτατο όριο παράδοσης των ειδοποιήσεων εντός 5 λεπτών.

Ανάκτηση Δεδομένων από το MinIO: Στόχος είναι η παροχή πρόσβασης στα ιστορικά δεδομένα θερμοκρασίας που αποθηκεύονται στο Minio μέσω του σημείου REST με αποδοτικότητα και ανταπόκριση εντός 3 δευτερολέπτων.

Παρακολούθηση Επίδοσης: Στόχος είναι η υλοποίηση εργαλείων παρακολούθησης απόδοσης, όπως τα Prometheus και Grafana, για τη διασφάλιση της υγείας του συστήματος και την παρακολούθηση της καθυστέρησης, με διατήρηση της υγείας του συστήματος εντός αποδεκτών παραμέτρων και παρακολούθηση της καθυστέρησης για βέλτιστη απόδοση.